Although the above description focused on notebook computer system power management, the present invention may be applied to desktop and home computer systems as well. Alternatively, the present invention may be used to power manage PCMCIA hardware and for thermal control of future generation CPU's.

Note that for Real Mode DOS or Standard Mode Windows environments, it may be appropriate to use POWER.EXE for real mode power management.

Thus, a method and apparatus for reducing power consumption in a computer system using virtual device drivers has been described.

I claim:

1. A computer system comprising:

a bus;

at least one memory coupled to the bus for storing data, including an operating system; and

a central processing unit (CPU) coupled to the bus running the operating system with a virtual device driver (VxD), wherein the virtual device driver performs device idle detection using one or more events timers indicating the activity level of at least one local device, and further wherein the virtual device driver places idle local devices in a reduced power consumption state when associated events timers indicate that no activity has occurred for a predetermined period of time.

2. The computer system defined in claim 1 wherein the virtual device driver performs system idle detection.

3. The computer system defined in claim 1 wherein the virtual device driver comprises I/O trapping capabilities to perform idle detection.

4. The computer system defined in claim 1 wherein the virtual device driver comprises a VxD trap handler to perform idle detection.

5. The computer system defined in claim 1 wherein the virtual device driver comprises a chained-interrupt trap handler to perform idle detection.

6. The computer system defined in claim 1 wherein the memory stores data structures indicating enabled local devices being monitored by the device driver.

7. The computer system defined in claim 1 wherein the memory stores data structures indicating events being monitored by the device driver.

8. The computer system defined in claim 1 wherein the memory stores data structures indicating I/O address ranges for local devices.

9. The computer system defined in claim 1 wherein the memory stores data structures indicating activity level of local devices to the device driver.

10. The computer system defined in claim 1 wherein the memory stores data structures indicating power management states into which the device driver may place the computer system.

11. A computer system comprising:

a bus;

a central processing unit (CPU) coupled to the bus running an operating system and at least one power-unaware application, wherein the operating system has a virtual device driver performing device idle detection using one or more events timers indicating the activity level of at least one local device, and further wherein the virtual device driver places idle local devices in a reduced power consumption state when associated events timers indicate that no activity has occurred for a predetermined period of time transparent to said at least one power-unaware application.

12. The computer system defined in claim 11 wherein the virtual device driver performs system idle detection.

13. The computer system defined in claim 11 wherein the virtual device driver comprises I/O trapping capabilities to perform idle detection.

14. The computer system defined in claim 11 wherein the virtual device driver comprises a VxD trap handler to perform idle detection.

15. The computer system defined in claim 11 wherein the virtual device driver comprises a chained-interrupt trap handler to perform idle detection.

16. The computer system defined in claim 11 wherein the memory stores data structures indicating enabled local devices being monitored by the virtual device driver.

17. The computer system defined in claim 11 wherein the memory stores data structures indicating events being monitored by the virtual device driver.

18. The computer system defined in claim 11 wherein the memory stores data structures indicating I/O address ranges for local devices.

19. The computer system defined in claim 11 wherein the memory stores data structures indicating activity level of local devices to the virtual device driver.

20. The computer system defined in claim 11 wherein the memory stores data structures indicating power management states into which the virtual device driver may place the computer system.

21. A computer system comprising:

at least one bus;

a memory coupled to said at least one bus;

a device coupled to said at least one bus;

a processor coupled to said at least one bus, wherein the processor is configured to execute a virtual device driver to control placement of said device into a reduced power consumption state.

22. The computer system defined in claim 21 wherein the virtual device driver performs idle detection for the device.

23. The computer system defined in claim 21 wherein the virtual device driver performs idle detection for the device using at least one event timer indicating the activity level of the device.

24. The computer system defined in claim 23 wherein the virtual device driver places the device in a reduced power consumption state when said at least one events timer indicates that no activity has occurred for a predetermined period of time.

25. The computer system defined in claim 21 wherein the processor runs at least one power-unaware application and the virtual device driver places the device in the reduced power consumption state transparent to said at least one power-unaware application.

26. The computer system defined in claim 21 wherein the memory stores data structures indicating enabled local devices being monitored by the virtual device driver, events being monitored by the virtual device driver, I/O address ranges for local devices, and activity level of local devices to the virtual device driver.

27. The computer system defined in claim 21 wherein the virtual device driver comprises a VxD trap handler to perform idle detection.

28. The computer system defined in claim 21 wherein the device comprises an I/O device.

29. A method for controlling an input/output (I/O) device, said method comprising the steps of:

executing a virtual device driver;

monitoring activity of the I/O device;

detecting the I/O device being inactive for a predetermined period of time; and

the virtual device driver placing the I/O device in a reduced power consumption state in response to the I/O device being detected as inactive.

30. The method defined in claim **29** further comprising the step of initializing, at boot-up time, a plurality of data structures associated with the virtual device driver.

31. The method defined in claim **29** wherein the step of monitoring comprises the virtual device driver monitoring activity of the I/O device at the occurrence of a system timer interrupt.

32. The method defined in claim **29** further comprising the step of varying the predetermined period of time.

33. The method defined in claim 32 wherein the predetermined period of time is varied based on desired power savings.

34. The method defined in claim 29 further comprising the step of the virtual device driver adjusting an events timer according to activity of the device.

35. The method defined in claim 29 further comprising the steps of:

a configuration manager notifying the virtual device driver of system resources being remapped; and

the virtual device driver examining its data structures to adapt itself to the remapped system resources.

* * * * *

36. An article comprising a plurality of computer instructions, which when executed by a computer system, causes the computer system to perform the steps of:

monitoring activity of a device;

detecting the device being inactive for a predetermined period of time;

placing the device in a reduced power consumption state in response to the device being detected as inactive; and

managing said device to allow more than one application to use the device.

37. The article of claim 36, wherein the plurality of instructions is a device driver.

38. The article of claim 37, wherein the device is an Input/Output device.

39. The article of claim 38, wherein execution of the plurality of instructions, causes the computer system to further perform the steps of:

checking a status of an event timer at regular intervals.

40. The article of claim 39, wherein the plurality of instructions includes at least one idle timer.

41. A method for managing devices within a computer system, said method comprising of:

monitoring activity of a device;

detecting the device being inactive for a predetermined period of time;

placing the device in a reduced power consumption state in response to the device being detected as inactive; and

managing the device to allow more than one application to use the device.

42. The method of claim 41, wherein the method is performed by a device driver.

43. The method of claim 42, wherein the device is an Input/Output device.

44. The method of claim 43, wherein the method further includes the step of:

checking a status of an event timer at regular intervals.

45. The method of claim 44, wherein the device driver includes at least one idle timer.

46. A computer system comprising:

a bus;

a memory coupled to the bus;

a first device coupled to the bus;

a processor coupled to the bus; and

a second device having stored thereon a set of instructions, which when executed by said processors, manage the first device to allow more than one application to use the first device, and monitor activity of the first device, and in response to said first device being inactive for a predetermined period of time, place the first device in a reduced power consumption state.

47. The computer system of claim 46, wherein the set of instructions stored on the second device is a device driver.

48. The computer system of claim 47, wherein the first device is an Input/Output device.

49. The computer system of claim 48, wherein the device driver checks a status of an event timer at regular intervals.

50. The computer system of claim 49, wherein the device driver includes at least one idle timer.